Background

In my individual analysis, I set out to answer the question of whether we can accurately predict a movie or television show's IMDB score using certain attributes of the production and the actors and directors who were involved in the production. I explored multiple linear regression as a statistical method to predict this metric.

Before I created my models, some issues in the data were fixed through data cleaning and wrangling. The first issue was with the categorical variables in the data that I wanted to use in my model, those being genre and production country. These attributes were originally presented as lists, but needed to be converted to a single value to include them in the regression model. I took the first element in each list to signify the most relevant genre and the most prominent country of the movie or TV show. Furthermore, I condensed the production country attribute into production continent, decreasing the potential values and simplifying the analysis.

In order to get the most insight out of the data collected, I split the data into movies and TV shows. The number of seasons attribute only related to the IMDB score of TV shows, while the number of Oscars won and the number of Oscar nominations only related to the IMDB score of movies. These columns would have been dropped if I used the entire dataset to create my model, so breaking it up allows me to maximize the value of the dataset.

Another issue present in the dataset was the age certification attribute. Nearly 50% of rows had a NULL value, but the attribute might have an impact on IMDB ratings for both movies and TV shows. As a result, I further split the data for movies and TV shows into two sets - one with the null rows dropped, and the other with the entire age certification attribute dropped.

Analysis

In total, I created linear regression models to predict the IMDB score for four datasets:

- Movies on Netflix with non-null values for the age certification attribute
- Movies on Netflix without the age certification attribute
- TV Shows on Netflix with non-null values for the age certification attribute
- TV Shows on Netflix without the age certification attribute

For each of these datasets, a 67/33 training/testing split was used for training and validating the models. Additional cleaning included removing irrelevant columns and NA values since linear regression cannot use rows with null values.

The dataset for movies with age certification values included 13 predictor variables and a single response variable, IMDB score. I created an initial regression model with this formula:

```
Unset
lm(imdb_score ~ release_year + age_certification + runtime + genres +
imdb_votes + noscars + nnons + production_continent + actormedian +
actorrange + directormean + directormedian + directorsonmovie,
data=movies_training_age)
```

To decrease the complexity of the model, I used the step() function to create the most optimal model based on its AIC metric.

```
Unset
movies_age_step_lm <- step(movies_with_age_model_lm, direction="both")
```

From the summary statistics of this model (not shown for brevity), I further pruned the model and only kept predictor variables that were significant at the 5% level. Before pruning, the step function gave me a model with an Adjusted R-squared value of 0.2852, and after pruning, the Adjusted R-squared value of the model decreased to 0.2781.

For the movie dataset without the age certification variable, the same procedure was used except that age_certification was removed from the initial lm() model creation. This new model was created with the other 12 predictor variables, still trying to predict IMDB score of movies.

```
Unset
lm(imdb_score ~ release_year + runtime + genres + imdb_votes + noscars +
nnons + production_continent + actormedian + actorrange + directormean +
directormedian + directorsonmovie, data=movies_training_no_age)

step(movies_without_age_model_lm, direction="both")
```

The model created by the step() function using “both” as the search direction had an Adjusted R-squared value of 0.205. After pruning the model to contain only significant variables, the Adjusted R-squared value was 0.202.

Creating models to predict IMDB scores of shows followed the same general procedure but with different predictor variables. The initial regression model consisted of nine predictor variables and was constructed using this formula:

```
Unset
lm(imdb_score ~ release_year + age_certification + runtime + genres +
seasons + imdb_votes + production_continent + actormedian + actorrange,
data=shows_training_age)
```

To decrease the complexity of this model, I again used the step() function with the direction of “both” to find the optimal model.

```
Unset
shows_age_step_lm <- step(shows_with_age_model_lm, direction="both")
```

From the summary statistics of this model (not shown for brevity), I further pruned the model and only kept predictor variables that were significant at the 5% level. Before pruning, the step function gave me a model with an Adjusted R-squared value of 0.2129, and after pruning, the Adjusted R-squared value of the model decreased to 0.1968.

For the TV shows dataset without the age certification variable, the same procedure was used except that age_certification was removed from the initial lm() model creation. This new model was created with the other eight predictor variables, still trying to predict IMDB score of TV shows.

```
Unset
lm(imdb_score ~ release_year + runtime + genres + seasons + imdb_votes +
production_continent + actormedian + actorrange,
data=shows_training_no_age)
```

```
step(shows_without_age_model_lm, direction="both")
```

The model created by the step() function using “both” as the search direction had an Adjusted R-squared value of 0.2349. After pruning the model to contain only significant variables, the Adjusted R-squared value was 0.2291.

Conclusions

All of these linear regression models have relatively low Adjusted R-squared values, which means that not much variance of IMDB score is explained by the predictor variables for both movies and TV shows. No model was much better in comparison to the other models at predicting IMDB scores for movies, and the same could be said for models predicting IMDB scores for TV shows.

To further explore the differences in the models, I calculated the root mean squared error between the IMDB scores contained in the test dataset and the predicted IMDB scores based on the model created. The results are summarized in the following tables:

RMSE for predicting IMDB score of movies	Model created from step() function	Pruned model with only significant predictor vars
Dataset with age_certification predictor	0.965	0.966
Dataset without Age_certification predictor	0.973	0.984

RMSE for predicting IMDB score of TV shows	Model created from step() function	Pruned model with only significant predictor vars
Dataset with age_certification predictor	0.942	0.953
Dataset without Age_certification predictor	0.977	0.980

```
Unset
# Example function call to calculate the RMSE of the model trying to predict
# IMDB scores of movies with the age certification attribute included

rmse(movies_test_age$imdb_score, predict(movies_age_step_lm, movies_test_age))
```

The results show that the RMSE for both models when predicting the same data are very similar. Typically, the model with the lowest RMSE is the best at predicting the response variable, which was the model created using the step() function with the age certification predictor. However, these models include multiple predictor variables that are not significant at the 5% level, which is not typical for a regression model.

One surprising thing was that age certification was not a significant predictor of movie IMDB rating, but the model created on the dataset that contained the predictor performed better than the model created on the dataset from which the column was dropped. For TV shows, certain age certifications were significant in the final model, which also performed better than the model that did not include the column as a predictor.

In general, the models had difficulty accurately predicting low IMDB scores - anything below a score of 5 - but this could be due to the dataset not containing many movies and shows with low ratings. Their relatively low Adjusted R-squared values mean that the linearity assumptions for multiple linear regression may be inadequate for predicting IMDB scores based on attributes of their production.

Even though linear regression may not be suitable for our question, I would choose the pruned models with the age certification attribute to predict the IMDB score of movies and TV shows. Although these models have slightly greater RMSE values compared to the unpruned models, it is more beneficial to choose the simpler model where all of the predictors are significant.

Appendix

release_year: The release year of the movie or TV show
age_certification: The official age certification of the movie or TV show
runtime: The length of an episode (for shows) or length of a movie
genres: The most relevant genre for the movie or TV show
production_continent: The continent where the majority of production took place
seasons: The number of seasons of TV shows only
imdb_score: The movie or TV shows score on IMDB.com
imdb_votes: The number of votes for the movie or TV show on IMDB.com
noscars: The number of Oscar awards won for movies only
nnons: The number of Oscar nominations for movies only
actormedian: The median number of occurrences for actors on the movie or TV show throughout all titles on Netlifx
actorrange: The range of occurrences for actors on the movie or TV show throughout all titles on Netflix
directormean: The average number of occurrences for directors on the movie or TV show throughout all titles on Netflix
directormedian: The median number of occurrences for directors on the movie or TV show throughout all titles on Netflix
directorsonmovie: The total number of directors on the movie